

Q U A N T A

THE NEWSLETTER OF  
THE INDEPENDENT QL USERS' GROUP

VOLUME 2 ISSUE 2

MARCH 1985

CONTENTS

1. Information on the Group
2. Editorial
2. Members' letters
16. Problem section
17. Beginners' section
21. Long strings
21. Using Talent's Zkul with disks
22. Book review - 'Desk-top Computing with the Sinclair QL for Professions and Businesses'. by Barre Miles
22. A graphics QL
24. The GST QC compiler
25. A simple program using the QC compiler
26. Croydon sub-group
26. Book review - "QL SuperBASIC - A Programmer's Guide". by John Wilson
26. Book review - "Introduction to Simulation Techniques on the Sinclair QL", by John Cochrane
27. Book review - "Exploring the Sinclair QL", by Andrew Nelson
28. Run-time Archive
28. New products
28. Bugs in the BASIC
29. OEL "doing a Sinclair"!
30. FORTH on the QL
31. Cambridge workshop
31. ZX Microfair
31. Psion Chess copy protection
32. Caps lock. on/off
32. A Quill merge program
33. How long is a string?
34. Quill Version 2 bus shock horror!
34. ALCC London Computer Fair
35. BASIC extensions
38. Tunes on the QL

### INFORMATION ON THE GROUP

Membership of IQLUG is by subscription to Quanta, the Group's newsletter, which is published monthly. Membership details are obtainable from the secretary. Membership of the group is open to anyone with an interest in the Sinclair QL microcomputer.

Members requiring assistance with problems related to the QL may write to, or phone the secretary. An attempt will be made to put them in touch with a member who can help with the problem.

Workshops will be arranged from time to time in various parts of the country.

The Group maintains a software library. Most of the programs are free to members.

A library list will be published from time to time. Programs are obtainable from the sub-librarians.

A list of members in any particular area is obtainable from the secretary.

Please send all contributions to the newsletter to the editor.

Chairman and  
Newsletter Editor

Leon Heller,  
8 Morris Walk,  
Newport Pagnell,  
Bucks. MK16 8QD.  
Tel: (0908) 613004

Secretary and  
Newsletter Publisher

Brian Pain,  
24, Oxford Street,  
Stony Stratford,  
Milton Keynes, MK11 1JU.  
Tel: (0908) 564271

Sinclair, QL, QDOS, QLUB and ZX Microdrive are trade marks of Sinclair Research Ltd.

Quill, Abacus, Archive and Easel are trade marks of Psion Ltd.

### NOTES FOR CONTRIBUTORS TO QUANTA

Contributions should be printed on A4 paper, using a fresh ribbon. Do not fold the sheets, but keep them flat when posting.

Quill files on Microdrive cartridges (cartridges will be returned) are acceptable, as are handwritten articles.

Short programs can be included, but lengthy programs should be placed in the library, and a description sent in for the newsletter.

### STOP PRESS

Will those who ordered their TOOLKIT from SINCLAIR Pre-February using their credit card, please re-order as we have been advised the original orders were not processed.

Brian Pain 19th March '85.

\*\*\*\*\*LIVERPOOL WORKSHOP Sunday 21st April '85\*\*\*\*\*

## EDITORIAL

Sinclair now seem to be marketing the QL properly, with several pages of advertising in all the magazines, publicising the third-party software and hardware available. I have heard a rumour that extensive TV advertising is about to start, also. It seems to be having the desired effect, since the QL has at last crept into the PCN top 10 list of machines under £1000, in 10th Place.

The Swindon workshop was a great success, with the hotel having to turn away over 20 people who wanted accommodation on the Saturday. The AGM was held on the Sunday, and Brian and I were re-elected unopposed. Chas Dillon was also elected to serve on the committee. It was agreed that the new constitution of the group would be circulated to all members for their comments. We hope to have it available with the next issue of Quanta.

Free QLUB membership in perpetuity is now offered to all new purchasers of the QL, but future software upgrades will have to be paid for.

Strathclyde University is to provide every student with a QL. We have sent them details of the Group as many of the students might be interested in joining.

GST and Metacomco are offering IQLUG members discounts of 10% and 20% respectively. You must send a current label from the envelope in which you receive Quanta, to prove that you are a member. The savings on one or two items of software might pay for your IQLUG subscription, so it is quite a good deal for you and the Group.

Some machines have slipped out with JS ROMs. This has a few extra features, such as error trapping, but some additional bugs, also. Apparently, the ROMs were supposed to be for test purposes, but some of them ended up in repaired machines, somehow. I have heard that an even later version is apparently being tested now, which should appear in new machines before too long. I don't know what Sinclair's policy will be about upgrading AH and JM machines, but I for one wouldn't mind paying £20 or £30 for it

Leon Heller

## MEMBERS' LETTERS

Members wishing to use CASSFILE (available from the library) with the new Archive (V.2) should note the following:-

- 1) The screen files 'cfs5', 'cfs6' and 'cfs7' must be converted to the new format using SCRCON\_BAS which will be found on the new Archive cartridge.
- 2) Search the two sets of procedures l('c' and 'ch') for occurrences of 'sload <filename>' immediately followed by 'sprint'. It is necessary to insert the command 'screen' between these two commands. Note that a command within a procedure does not always have the same effect as a command entered from the keyboard.

I can confirm the value of the Simplex 256K RAM, but I would like to know more about the way it is organised. The 256K X 1 chips have only 16 pins, so there must be some form of address multiplexing, yet the RAM still seems very fast, as mentioned in your report.

I would also like Quanta to deal with the 1/0 addresses from \$18000. I can look at these. but I have no idea what they mean or how they are physically located.

Leslie G. Hammerton,  
3 Piercefield Road,  
Freshfield,  
Formby,  
Merseyside L37 7DG.

/\* . All modern dynamic RAM chips use multiplexed address signals. to reduce the number of pins. The multiplexing and demultiplexing is so fast that it can easily keep up with the microprocessor. so there is no performance deficit. at least as far as the QL is concerned. The TMS4500 on the Simplex card can handle the multiplexing for 64K RAMs and there is an additional 74LS157 to perform the extra multiplexing needed by the 256K devices used. Texas Instruments will shortly bring out a bigger version of the TMS4500 that will handle RAMs up to 1M bit, I am told! \*/

---

I found your article on the Camberley visit very interesting.

Sinclair appear to have not just pulled their socks up, but to have several new pairs.

My QL developed a fault on Sunday evening. 24 February. I returned it the next day. Yesterday, 5 March, i.e. eight days from dispatch, I received a brand new QL plus a second set of Psion programs, and a brief standard note saying that 'the fault wasn't immediately repairable, could be something random that had developed, something I'd programmed, or a peripheral. If it repeated on the new QL please don't send it back. phone us first.'

Me cartridges runneth over. I cannot see myself purchasing more for a very long time.

Wil Jackson,  
131 Denton,  
Malden Crescent,  
London NW1.

Quanta is getting better and better. As a beginner there is usually a Jewel or two to pick out. In the Jan. issue John Roberts and his BEEP and Windows. etc. and Andrew Rigsby. Page 31.

I was very excited by the Calendar program on. page 23 as I am editing letters written in '39-'92. and need it to know days of the week to identify when letters were written - but look what the program gave me for 1985. Can we have one that works, please?

Peter Tatton-Brown,  
Grasspark,  
Brayford,  
Barnstable.  
E32 7PX.

I feel that I must take exception to the letter published in QUANTA issue number 11, under the heading 'RECURSIVE FUNCTIONS ON THE QL. Mr. Fauvel states in that article, on page 23, "But there can be a problem, even when the numbers are simple and manageable". Mr. Fauvel is referring to a Fibonacci series programmed as a recursive function. To state that the numbers in his recursive algorithm are simple and manageable is a cross misstatement. I believe that the algorithm "APPEARS" to be simple and manageable, but upon close examination we see that the programming method used causes the FUNction itself to be called many times over. The overhead to handle a FUNction in SuperBASIC is really very small. It is the method used by the algorithm which is at fault.

The following calculation should help to clarify the matter:-

- 1) When calling the FUNction, we perform:-  
Fibonacci (number)  
if n = 1 : return 1  
if n = 2 : return 1  
return Fibonacci(n-1) + Fibonacci(n-2)  
On average, I think we can say 4 SuperBASIC statements per call.
- 2) The question is, how many times do we call the FUNction in order to perform any calculation:-  
This is found to be as follows:-

Rank in series.	Calls to FUNction
1	1
2	1
3	3
4	5
5	9
6	15
7	25
8	41
9	67
10	109
11	177
12	287
...	...
...	...
31	2692537
32	4356617

Mr. Fauvel was interested in the 31st Fibonacci number and therefore request SuperBASIC to execute the FUNction more than 2.6. MILLION times and then complains that the QL takes a long time. ANY computer so programmed would take a long time, John. Perhaps You would care to use the FUNction listed below:-

```
100 INPUT "Enter number : ";number
110 PRINT Fibonacci (number)
120 STOP
130 DEFine FUNction Fibonacci (n)
140 LOCAL a, b, c, i
150 IF n = 1 : RETURN 1
160 IF n = 2 : RETURN 1
170 a = 1 : b = 1
180 FOR i = 3 TO n
190 c = a + b
200 a = b : b = c
210 NEXT i
220 RETURN b
230 END DEFine Fibonacci
```

I think you will agree, by no means as elegant as John's FUNCTION by a great deal easier on the QL. I believe that there is a great deal to criticise on the QL, including some of the ways in which SuperBASIC has been implemented, but this is not one of them.

**"AND NOW FOR SOMETHING COMPLETELY DIFFERENT"**

One of your listeners, Mr. Giles Powell, had a problem with listing a program to a printer with only the trailing line feeds going out. I have found the following little program, coded into your own will do the trick. This will operate only so long as your program has been "RENUM"ed correctly:-

```
100 Program code in here
.....
32760 OPEN #3,ser1ehr           Open the printer device.
32761 FOR n = 10 TO 10 high_line STEP 10 high_line must be set
32762 LIST #3,n                 Print a Listing line
32763 PRINT #3,CHR$(13)         Add a carriage return
32764 NEXT I                    Continue Looping
32765 CLOSE #3                  Shut down the Printer
```

Keep up the good work, I listen to your Program every week on VHF and especially like the girl with the green dress.

Ron Dwight,  
T:m1 Softronic  
Sufikuja 3B14  
02120 Espoo 12  
Finland.

As a new member of IQLUG may I say how useful the back issues of the QUANTA have been.

I felt that it would be of help to some members to relate my problems in setting up my typewriter to be used as a printer for the QL. The typewriter is a SILVER REED EX43N and it uses an interface common to both this and the SILVER REED EX44. This is interface I/F40.

Setting up the interface is easy with lots of choices available within the PSION driver and the interface. Unfortunately the interface although supposedly supporting a baud rate of 9600 often seems to lose some of the initial data sent to it. After spending many hours trying alternative printer initialisation codes, I tried a different baud rate and avoided odd characters at the start of a printing session.

My choice for the serial interface is:-

```
FOUR POLE DIPSWITCH  1.....ON
                     2.....ON
                     3.....OFF
                     4.....OFF
```

(This means English character code and CR+LF)

```
EIGHT POLE DIPSWITCH 1.....ON (Xon/Xoff)
                     2.....ON
                     3.....OFF (No-parity)
                     4.....OFF (8 bit)
                     5.....OFF (2 bit stop)
                     6.....OFF
                     7.....OFF
                     8.....OFF (6,7,8 = baud 1200)
```

INSTALL\_BAS Parity-none; baud rate 1200; continuous forms - no end of Line - <CR>,<LF>; translate 1 - "£,"# ; translate 2 - "#,92

The second item which I have experience with is the SANYO CRT-36 monochrome monitor from DIXONS designed for the SANYO MBC-550. It has an 18 MHz bandwidth, can display 85 columns

gives superb results. It costs about £130 although we were given a discount for buying it at the same time as a QL for the department I work in. Incidentally my own QL was bought at the end of November from BOOTS, has all the latest PSION program versions (ADB) but has an AH ROM (& D-09 number stamped on base). As all the software loaded first time, in contrast to someone I know who bought JM version from Smiths at the same time and is still waiting for a loadable QUILL, I kept the machine. Subsequent commercial programs have all loaded first time (PSION CHESS & TALENT ZKUL). The former is superb graphically and is a very strong and aggressive player while the latter is very entertaining and is also very well written without annoying spelling mistakes and grammatical errors. Both are worth every penny.

Dr. W. J. Fuggle,  
20 Widney Avenue,  
Selly Oak,  
Birmingham B29 6QE.

I received yesterday my second versions of the bundled software, and this is being written using Quill version 2.00. I cannot enthuse enough about it when comparing it with version 1.03. Not only is the whole programme resident in memory, but it does not access mdv2\_ for Def\_doc every other moment. The scratch pad can be held in mdv2\_ and regular documents which are required to be merged or loaded in standard form, such as the address header, can be held in mdv1\_ and merged or loaded as required so long as the "help" menu is not required. Different printer settings can be held on different tapes so that the type face can be interchanged between letters. There are two minor problems that I have so far located.

1) When copying you are in fact asked whether you want to keep or delete the passage being copied before you move to the new position. This is incorrectly stated in the body of the new sections of the manual, but correctly stated in the function keys explanation. You can however move the cursor key forwards and backwards, up and down lines and from word to word in both directions, provided you don't go back past the start, so that delineating the passage to be copied is much more simple. If you are used to the old form there is one annoying difference. Before, if you pressed <ENTER> it defaulted to keeping the passage to be copied, and you pressed 'd' to delete. Now you press 'k' to keep and ,<ENTER> to delete.

2) The other problem is equally minor. The last line of a document, whether it is a footer or the last line, is not printed, although the line is sent to the buffer. If you want it printed you have to use the printer's controls to form feed. It is then printed before the form feed is performed.

I have not had an opportunity to get to grips with the new Archive or Abacus, although they appear to be much improved, and Easel still leaves me mystified.

W.D.R. Spens,  
Marsh Mills Cottage,  
Over Stowey,  
Bridgewater,  
Somerset TA5 1HG

After the first steps in programming my QL, I am quite content with it, but there are some facts, that seems very disappointing to me.

Microdrive Access time is far too slow. On my spectrum I could load a 50 K Bytes Program within 10 seconds (never longer). And it would load and save with all variables.

In fact, QL Microdrives seem to imitate a floppy. Best way to get around this could be the saving of programs (with or without it's arrays). Surely there are system variables that tell where the program and where the variables are stored. But which variables are the ones?

The way the QL uses Microdrives is the main reason for the unreliability of the cartridges: QL uses Microdrives is not adequate to the device. There is a great difference in reliability whether the tape in run with long times with every access or the run time is as short as possible. In 1 year using Microdrives on the Spectrum I lost one file and no cartridge has been corrupted. In one month with the QL, I lost 3 cartridges (new ones), and the report "bad or changed Medium" seems to be a decimal counter. (Try ten times (once you get this message when you've longer files). It is a pity, to see how the abilities of the QL get lost only because all tend to make the Microdrive do something like a "mini-floppy" It will never be, But It could be a very fast "super tape". Why sacrifice this feature for an ideological "mini floppy"?

The other weak point is the way how the keyboard reacts on double strokes (repeating of the last stroke). Not so annoying: Shifted space bar has no effect. For typing even this must be corrected.

At last, the programs are a bad joke. I don't know how it was possible to make such long programs, that do work so slow and are full of bugs. Most of all, they are as well not proper for the storage-medium. The Microdrive can be very fast for loading and saving whole files, but they are far too slow for short files being stored subsequently.

Word processing with the QL. should be done in memory not on cartridge. The same is to say about Archive

Considering the small memory of the old Spectrum, I wonder how Programs like 'tasword' and 'masterfile' work (you have only 40K for program and data). They offer more and faster features. With 'tasword' I can have over 5 pages in memory and can read through them without of any Microdrive access.

If the "sedit" program of the group would only provide word wrapping, it was far more better than Quill.

I really hope for some programs that will make the QL work like it could. Or does Sinclair leave it all to the user - in this case, QL was a good micro for freaks but not for small business.

Wolfgang Goller,  
Rosenstr. 21,  
8105 Regensburg,  
Switzerland.

/\* I agree with your comment about the <shift> <space> not registering, I find it very irritating. I feel that your other problems are due, in part, to a faulty machine, as Microdrive operation on my machine, and that of most members, is quite reliable. \*/



I have just received my updated software and this is one of my first efforts with the new QUILL.

I must say I am delighted!!

The difference is quite amazing and to my mind seems to have solved my only complaint with the previous version - it was too slow. I suppose the knockers will be out in force again and find something to complain about, what on earth do people expect for £400?

Over the last nine months I must admit I have had some problems with my QLs but I think you should know that I have had a great deal of help from several of Sinclair's departments and in particular from "she who must not be named"!!!

The extension to our QLUB membership, nearly makes the £35.00 entry fee worthwhile. I wonder how many updates they will sell at £50.00 a set? If you were using the QL for a lot of word processing then I suppose £15.00 spent on the new QUILL would be money well spent.

Thank's for all your hard work on the IQLUG.

Derek Bell,  
Westfield House,  
Swannells Wood,  
Studham,  
Beds.

In my letter (Quanta 11) I said I would report back on whether a postage refund appeared. Well the QL was returned by my postman on a SUNDAY!! But no refund came. However I was given a new set of the Psion software. After application of small pieces of Sellotape, I had £19 worth of cartridges. A good enough refund on £5 of postage.

QLUB membership looks a bit better now! A new set of cartridges for £35 membership fee, instead of £50, though the expanded services are for all QL owners it seems, since I received letters intended for both QLUB and non QLUB members (who had sent in their guarantee registration cards.)

J.R.Catt's (Quanta 1.12) suggestion, that QUILL users redefine the copyright sign as ESC, is quite perfect and deserved a page to itself. Having seen it, trivial as it is, we all must have kicked ourselves for not having thought of it before. Of such simple, but infrequently offered ideas, is genius born. Thank you very much J.R.Catt.

Richard Kingslake (Quanta 2.1) refers to the restriction of a 128 bytes buffer in INPUT. Well it has to be something and a larger buffer may slow down the input of Just a few characters. The bypass is to use INKEY\$ instead, like this:-

```
OPEN IN #5, mdv_.....
REPEAT dataread
  IF EOF(#5) : EXIT dataread
  char$ = INKEY$(#5,-1)      n.b. don't leave this -1 out else the buffer
                             keeps on pushing data out regardlessly.
  Actions that process char$
END REPEAT dataread
```

It's not as quick as INPUT, but not that much slower.

Ian Cowan (Quanta 2.1) is unhappy about W.H.Smith's copying of PSION QL packages. However these surely run on QL computers only and PSION have a bundling agreement with Sinclair and will be paid according to the number of QLs sold. So the action Ian saw did nothing more than reduce PSION's duplication costs a bit and increase the service to some customers of W.H. Smith. Hence no losers

The important test will be how many copies of Version 2 PSION software get into the hands of version 1 owners without the exchange of £50.

R.C.Armstrong (Quanta 2.1) asks about a trace program for SuperBAS1C. Andrew Pennell published such a routine in Popular Computing Weekly Vol 3 No. 35 (30 Aug - 5 Sep 1984 ). It doesn't single step, but the speed is variable. I haven't tried out this routine, but Andrew does say that it gave odd results at times.

M.G.Bite (Quanta 2.1) asks about the ALT key. Well this quite a common key on computers. The terminal to my £4,000,000 IBM has one too!

On the QL it is used to give another value to each key. As described at the end of the CHARACTER SET AND KEYS topic of the CONCEPTS section, when you press ALT and another key at the same time, then two codes are sent to the QL, FF followed by the code of the other key. An example of why this useful is in Tony Tabby's text editor SEDIT\_BAS on the LIBRARY 1 cartridge. In this you may be typing away as I am right now and want to save. All I do is hit ALT followed by 's'. SEDIT then sees the FF, recognises that a command will follow and then looks for the next character, in this case 's', so it saves this text. Hitting 's' alone just prints it. ALT in this case allows meaningful commands, in other cases it can give you F6 F7 F8 F9 F10.

Bill Cowhig (Quanta 2.1) queries the strange preference for 'l', 'j', 'k', 'i' in program listings. Well when the world was young and FORTRAN was king, variables starting I, J, K, L, M, N, were defined automatically as integers and most suitable for D0 ( =FOR) counters. The wealth of experience about computing and how to write programs was built up in this period, so now even among people who have never seen FORTRAN, the idea of I, J, K, L, M, N, as loop counters is part of the received wisdom about how to write code. Those FORTRAN programmers had, and still have, I, J, K, etc. printed in capitals on impact line printers. But on our QLs we are too lazy to hit the shift key and so get l, j, k, i, etc. and print the result out on comparatively cheap matrix printers. If we have to have two key presses why not let it be SHIFT i and let us have I and not dd, which looks like a small beer!

Quanta has many letters in it criticising the after sales service of Sinclair. Though I have suffered myself, is this fair? What free after sales service do you expect to receive for a £400 video or £600 3 piece suite? Are we not expecting the kind of free service that we hope to get for a £4000 car, but end up having to pay for?

Recent issues of the computer press have referred to Sinclair having to suspend deliveries of the QL because of excess stocks. Whenever they are/were, it wasn't at W.H.Smith, whose Sloane Square computer salesman told me we're selling all the QL's they could get and whose Ealing branch is out of stock. So the future looks good, since we expect good sales to result in good software.

The Consumers Association magazine "WHICH?" for March, carries a review of the QL in their reassessment of the top computers. The QL rates as 'OK but not outstanding'. Top 3 are the BBC, Electron and Spectrum, with BBC as overall top. Their down marking of our world beater is apparently due to the lack of independent software, though the PSION packages, especially QUILL don't come off too well..

I have spoken with John Cook, software editor of Popular Computing Weekly about putting programs into the IQLUG library. He said that if they published my listing they acquired the publication copyright. The distribution rights are different, and though he would

prefer a few weeks gap between publication in PCW and appearance in the library, such appearance is quite legal. What I didn't ask him about was the status of listings where the author does not offer to place them in the IQLUG library. Clearly there are many published programs which IQLUG members would like in the library. Does anyone know the status of such programs?

Gerard Phelan,  
17 Gunnersbury Court,  
Bollo Lane,  
London W3 8JL  
01-993-3273

You asked recently +or reasons why the free upgrades 04 Software should be universally issued rather than just to the SLOB members and I did start this letter to you in February to give some reasons but I am afraid it has lean dormant in a cartridge since then. I could give you the simple answer in the case of Quill that it is to atrocious that it must be re-issued universally, but i will expand on that subject in a moment. As far as the other packages are concerned I have not used them nearly as fully and therefore I have not experienced much difficulty; perhaps someone else can give chapter and verse on them. I was prompted to take up my quill again after receiving the latest issue of Quanta because I was astonished that you have not pursued the objective of free software upgrades for all owners of QLs. I personally \*am not affected by your decision since ~I am a member of QLUB and therefore receive my updates 'free'(about the only thing of worth!), but I can quite categorically state that I would have insisted on a refund of my money solely on the complete inadequacy of the Quill software had I not known that a reissue was in the pipeline. I chose the QL principally because of the packaged software, (if there had been no software I doubt whether I would have made the same choice), my main interest in the software is 'Quill' which I try to use quite extensively in a retirement activity and I am absolutely certain that anyone with less time than I have would have had to give up in disgust. For instance I recently heavily amended a seven page document which took a long time and which ended up with the loss of the amendments and the loss of the complete cartridge. As far as the original version 1.03 of QUILL is concerned I reckon that only about five percent of the time that I have spent on it has been productive,(Although some of the blame has, perhaps, to be allocated to other causes such as cartridges being pushed out of drives.) and that it is obvious to me that the product does not in any way live up to Sinclair's claims. As a result of my experiences I can say for instance that with version 1.03 I never use Tabs, Margins, Header or Footer, that I only use monitor mode and 80 column width,that I 'save' about every 30 lines, that I maintain at least four copies of each document, that there has been about a 50 percent chance of the hard copy being corrupted, that whole cartridges are occasionally corrupted and lost for no apparent reason, that the keyboard locks out for no apparent reason in the middle of documents, and that the symptoms can vary using different copies of QUILL tapes.

To provide evidence I can only quote from the log which I keep, which is about the size of a school exercise book, and which is almost full of fault records. Here are a few:

Corrupted print-out. This is devastating and commonplace and as admitted by Psion to be a problem. The symptoms are twofold, in one case the text can be printed out in one long string (Headings, paragraphs, insets, tabulations all disappear.), In the other cue the fault inserts line or form feeds (In one case a single page got spread over 17 pages.) The fault is not transient it is in the saved file, And worst of all it is also in the copies.

'Import' fails regularly giving error- message 'File i/o incomplete', and the only way out is to 'Reset' because the keyboard is locked out.

Error message 'Not a valid Quill file' is returned regularly, although the file shows in the directory and perhaps loads on a subsequent attempt.

The page number sometimes appears at both the bottom and top of a Page

'Fatal Error' message occurs at crucial moments for no apparent reason and the only way out is to reset.

Cloning seems to be a matter of luck as to whether or not one can get past the trap of "bad or changed medium".

The error message 'Unable to open file on mdv2. Press any key to continue', does not work, the only escape is to 'Reset'

'Copy' occasionally does quite dreadful things to the text.

The Cursor seems to develop a will of its own at times and often the only solution appears to be to delete a line or two of text, or in the worst cases to 'Reset'.

I think this is more than enough. Individually some of the items are perhaps not of great significance but taken as a whole it means that something is constantly going wrong with the attendant frustration and loss of time, and very often the loss of text. However, the real 'killer' is the corruption of the file when trying to obtain hard copy and in order to try to minimise the effect I have been writing one page at a time and saving it individually. Very tedious and very unsatisfactory!

K.P.A. Dyer,  
Kineff House,  
Devon Bank,  
Guildford,  
Surrey GU2 5DQ.

/\* I still maintain that most of our members seem quite happy with the original versions of the Psion software. Some members seem to have had similar problems to yours. but when I tested my original version of Quill, I experienced nothing like the number of bugs that you found. Although the ones I found were very serious. It could be that somehow your version of Quill has been corrupted in some way - did you return it to Sinclair for replacement? One member told me that he has written a 10.000 word thesis using the original Quill with no problems! \*/

As I suppose every new member to a club must be, I am very enthusiastic about the whole thing, and I was most impressed with your (or Brian's) speedy reply to my letters and how quickly you had sent me the requested backnumbers of QUANTA, even though you did not send them by registered mail as I had requested - what ever happened to the £4 I had sent just for this purpose? (No, I don't want them back, use them for the library cartridge I had mentioned above).

Still on the subject of IQLUG, would you be interested at all in me advertising our club around here? I even considered back-page advertising IQLUG (out of my own pocket, of course, in one of the national computer magazines here in France. I shall only do so however if you think this worthwhile. I could be something like your local

antenna in Paris (or in France for that matter). Do you have any other member here in the Paris area ? If so, could you publish my name and address in QUANTA as I would like to get into touch with them and have a chat about our computer. I propose my services as a sub-librarian for the Paris region (or for Franca if you wish).

I think that what I shall miss most is not being able to visit or attend any of the club's workshops or meetings. In this respect I have a small comment to make: I received today (Feb 19)my 'January' copy of QUANTA. In it you are drawing our attention on the London workshop on January 27 (!).Did anyone turn up? I know I can't be expected to come to the workshops but still, telling me about it nearly a month after it happened is not very nice...

### **SINCLAIR**

I have been having some trouble with Sinclair. (Who hasn't ?) They won't let me join the "Qlub" to which I would like to adhere in order to get the Psion packages updated. I have sent them a rather unsatisfied letter. I have now received a reply stating the same all over again i.e. me being an 'overseas' customer, them not exporting the QL by mail order and all that. This will create them quite a negative publicity over here - I have half a mind to make sure of that. All other 'overseas' customers be warned!

### **PRINTER CONNECTION**

I don't intend to write yet another story of how I got my particular printer (a NEC P2, with which I am very satisfied) to talk to my QL - or rather vice-versa. I would rather give some sort of general advice on which line, to be found on the printer's side of the connection, does what; that way most people will be able to figure out for themselves which line they need to connect or not. The lines can be divided into two groups: Those that go from the computer to the printer and signal that something is coming and those that go from the printer to the computer, telling the latter that it is ready to receive the data etc. The lines most likely to be encountered are as follows:

#### **A - Lines from the computer to the printer**

Carrier Detect (CD): Normally used with a modem, it shows the printer that the computer is there - that's all.

Data Set Ready (DSR): Shows that the computer is operative. These two lines should be connected to the QL's +12 Volt line i.e line number 6, which is the orange one, when one opens the printer cable supplied by Sinclair.

Clear To Send (CTS): Shows the printer that the computer is about to send it the data - should be connected to the QL's CTS line (the red one).

Received Data (RxD): Via this line the data is actually transmitted to the printer.

#### **B - Lines from the Printer to the computer**

Data Terminal Ready (DTR): Shows the computer that the printer is ready to accept the data. The state of this line changes, for example, if the printer buffer is full, thus ordering the computer to stop transmitting for a while. This line should be connected to the computer's DTR line (the blue line).

Request To Send ( RTS): Has the same function as DTR and should be left unconnected.

Transmitted Data (TXD): If the printer could send data to the computer, It would be via this line.

Reverse Channel (RC): This line can sometimes be found. It has the functions described for the DTP line. Generally it can be made to function via the DTR line on the printer interface.

Wolfgang Lenerz,  
5, Passage Tiere ,  
75011 Paris,  
FRANCE.

/\* All members in the London area were sent details of the London Workshop separately, and the meeting was very well attended. We also mentioned It in Quanta so that If anyone outside London wanted to come along, and there were some places left, they could attend the meeting. The same applies to the Cambridge meeting mentioned In this issue - all members within easy travelling .distance from Cambridge will have received details separately. \*/

Thank you for Vol. 2 No 1 of QUANTA. In the same post. on 15<sup>th</sup>. March, I received the updated (version 2.00) Psion software - Quill, Abacus, Archive, and Easel.

The Improvement in loading time is the first thing which struck me. The average time for loading, from pressing F1 or F2 to being ready for input, is about 20 seconds. Starting-up time is improved by the removal of the intermediate file START; in version 1 this was first loaded into the Resident Procedure area and then CALLED to load in the main program. In version 2, the main program is called by EXEC W directly from the BASIC loader. Another Improvement is the increased speed of cursor movement in Quill. There is less tendency for the cursor to disappear at odd moments.

Archive now has a full range of standard arithmetic functions added (sin(), cos(), etc)

All the programs benefit from a reduced code size, allowing more space for data and needing no overlays. The free space in Easel is now 12K, in Abacus 23K, and in Archive 20K.

The printer drivers for Quill, Abacus and Archive have been modified to permit use of a port other than SER1. Up to 10 characters may be re-defined by TRANSLATE. This means that the printer drivers configured by INSTALL\_BAS for version 1 are incompatible with Version 2; a basic program CONVERT\_BAS is provided for conversion.

Screen files for Archive must also be translated. using SCRCON\_BAS. I had less luck with this It appears that if the screen is designed for 80 characters, so that some of the variables cross column 64, then when the screen is displayed in 64-character mode the variables wrap round to the start of the same line instead of the next one as previously. This resulted in strange blank areas on my screen layout until I realised what was happening and re-designed the screen

A choice of three printer-drivers are provided for Easel; the default one is Epson FX-80 or similar (stored as GPRINT\_PRT), but files are provided for an Integrex colour printer (INTGX132\_PRT) and Oki (?) (JX80\_PRT).

To make the printer driver work correctly with my Brother HR-5, I had to find and change the code which sets the vertical print spacing. The FX-80 driver uses ESC "A" 8 to set the spacing to 8/64 inch. I changed this to ESC "A" 4 to select 4/32 inch for the HR-5. The following program preserves the FX-80 driver and makes a new HR-5 driver;

```
100 COPY MDV1_GPRINT_PRT TO MDV1_FX80_PRT
110 DELETE MDV1_GPRINT_PRT
120 buffer = RESPR(1024)
130 LBYTES MDV1_FX80_PRT, buffer
140 POKE buffer + 410,4
150 SBYTES MDV1_HR5_PRT, buffer, 520
160 COPY MDV1_HR5_PRT TO MDV1_GPRINT_PRT
```

(Lines 110 and 160 are actually not essential, because any of the drivers on any storage medium may be temporarily installed from within the Easel Print command).

One further improvement; there is a program CONFIG\_BAS on the Abacus cartridge which can re-configure any of the four programs, and alter the default devices for program, data, and help files.

On matters brought up by contributors to QUANTA Vol 2 No 1:-

Richard Kingslake could get round his 128-byte buffer problem in reading strings from Microdrive files by reading them one character at a time, using INKEY\$. When defining console windows, of course, any size of keyboard buffer can be defined by the last parameter.

With Wilson Stothers, I have reservations about the software packages sold by Computer One. I have purchased their Pascal system. It works well. in general, except that, being a p-code interpreted system it is not as fast as I expected, and there are a few bugs. For Instance, writing a floating-point expression with a value of zero to a Microdrive file causes some stack imbalance, so that for-loops are terminated early, etc I spent a lot of time tracing that one. Otherwise, all the functions and procedures which Computer One include in their Pascal handbook seem to be there.

For my own system expansion, I now await the availability of the .CST disk interface without disk drives.

I already have two 40-track drives in older obsolete equipment, so I don't want Initially to spend more money purchasing new ones.

Frank Burgum,  
62 Caithness Road,  
Mitcham,  
Surrey. CR4 2EU.

I've just received my new set of the Psion programs and first Impressions are favourable. with the exception that the Archive tape refuses to recognise either "Backup" or "Format". But of course I'm not writing just to tell you that!

As I was rather critical of the printer driver program in Quill in a previous letter (Jan.p 20) I am bound to acknowledge that the new versions allow me to do almost anything that my printer can manage. For a start Quill now recognises any (if not all) of the extra characters hidden away in the QL. The driver allows ten translation functions and so its possible to have a fair selection of accented letters or other special signs at ones disposal. And thirdly the driver can be persuaded to backspace and thus produce combination characters though this is likely to upset the line justification since the backspace does not appear on the screen. It is also pleasing to see that Psion now use the method I explained in my letter for dealing with the £ and # signs.

Here is a sample of some characters which I've included in the driver:-  
à é ç ä è ú Å § and that is exactly how they look on my screen.

The backspace facility shows on the screen as the two characters separated by a left arrow which my printer can't reproduce so I have to cheat on that by using the backspace to make it from a and a <. Here is the screen picture:-

A←^ Y←= c←: <← will print as â ¥ c <

The backspace is called by typing Control \ and the code for the translation is 188.8.

Capt E. D. Fieldson,  
Miraflores,  
Troutstream Way,  
Loudwater,  
Rickmansworth ,  
Herts. WD3 4LA

### PROBLEM SECTION

I agree that the QL Advanced User Guide which was reviewed in Quanta Issue 12 is a very good book. but I have one question to ask: has anyone, following the instructions given in the book, been able to get the QDOS Experimenter program to work? Using TRAP 1 with D0=0 gives my QDOS version as 1.03., but no subsequent examples can be entered.

Len Leyrer.  
13 Castle Hill Avenue,  
Folkstone,  
Kent. CT20 2TD

-----

1) Routines callable from SuperBASIC to improve speed, flexibility and smoothness of animated graphics. In particular:-

A) Fast smooth move of an object about screen (panning a window is inadequate), and preferably...

B) Against and over a pre-drawn background, so that as an object moves, the original background reappears unchanged by the passing of the object. (My present attempts carve a blank swathe through the background).



C) A way of defining an object (or at least a rectangular area of image) to be tucked away in memory so that it can be retrieved as needed and plonked quickly back anywhere on the screen (redrawing from scratch is tedious and slow), preferably graphics coordinate-based, not Pixel-based.

2) A facility for programming the keys (at least the function keys, preferably any key combination) so that they can be used for automating repetitive tasks at QDDS level similar to "Pro-key" program for the IBM PC). Key assignments to remain resident once loaded.

3) Any games which really use the QL potential (as opposed to ones transcribed from less able micros).

4) Any slick software for "free-hand" drawing on the screen - Preferably with ability to save the object created as in 1C above. Input of drawing instructions in this case could be via the keyboard.

5) Any fairly cheap hardware to do the same by drawing on a pad or similar.

Dave Thomas,  
North Bandonie,  
Midmar,  
Aberdeenshire.  
Tel: (03303) 516 (home) (0224) 574588 (work)

/\* Tony Tebby tells me that the QL was designed in such a way as to prevent people programming the function keys, apparently in the interests of compatibility. It can be done, however, according to Ands Pennell, and the technique might be published before long.

Talent Computing Systems are selling what looks like a very nice drawing program called "GraphiQL". They are on (041-552 2128). \*/

-----

### BEGINNERS' SECTION

#### DATA INPUT :

Copyright Frank Hope 1985 (not to be reprinted without consent from the author).

If you are just starting to program in SuperBASIC, and have no prior knowledge of computers, I recommend you to build up a library of procedures, so that you need never begin a new program from scratch. You simply merge the necessary procedures and link them together. The procedures don't have to be original. you can take useful ones from programs printed in magazines, or even books, but as you gain experience you will probably start to produce your own.

There are a few points worth mentioning about programming. Firstly, you should always use different line numbers for each procedure in your library, so that when merged you will know where they are, and that they will not destroy other parts of the program. Secondly. don't be afraid of using GOTO's and GOSUBS. It is far better to get something to work than to spend hours producing well structured rubbish, (the structuring will come with experience). Thirdly. write the program for yourself, not others. Unless you intend to publish your work, don't bother about making the code easily read, just make sure that when you go back to it at a later date you know what everything is.

If you look at the QL user manual you will find 2 Keywords which allow information to be typed into the computer. These are, INPUT, and INKEY\$. If you are a beginner to programming, and are reading the manual for the hundredth time, the descriptions will probably mean absolutely nothing. The problem will be compounded when you try to use the words in a program.

When using INKEY\$ in it's basic form you will only be able to type 1 letter or 1 number. This is fine when you use a menu with less than say 9 options, but when you wish to type complex information into the computer it appears useless. The Keyword INPUT, however, does allow you to type in a line of information. The problem with it is that you can't control the type, or amount, of information entered. For example, you may design a beautiful screen. and use the following to allow numeric data to be entered :-

```
10 INPUT A: REMember enter a number
```

You then get some Wally who comes along and types his name, (string) and presses enter. Your beautiful screen could very quickly disintegrate into a horrible mess. What beats me about it all is that authors of so called, Advanced Programming Books, use INPUT, as above, in their program examples.

The procedures outlined below will allow you to control where on the screen your information is to be displayed, the type of input i.e. numeric/alphanumeric. and the number of characters to be typed. The procedures are used together to form the overall input routine. Don't worry if at first you can't understand how they work. Just put them into your library and use them when necessary. When you become more proficient at programming you will probably find improvements.

The first procedure entitled CUR, allows you to place the cursor anywhere on the current default screen. If you try to print something outside the maximum screen co-ordinates an error will be generated. To position the cursor at co-ordinates 50.50 you simply call the procedure by typing CUR 50.50

```
5000 DEF PROC CUR(X,Y):CURSOR X,Y:END DEFINE
```

The second procedure is the main bulk of the routine. Don't worry if at first sight it looks complex. You will find its features as and when you use it.

```
6000 DEF PROC INFO(W%.L%. LIMIT%)
```

The % sign tells the computer to use integers, this helps speed up execution. W% and L% state where you want the printing to be positioned, if you wish, you can change them to be more descriptive. e.g. row%.col%. Limit% indicates the maximum number of characters to be typed.

```
6005 a1%=L%:a%=0:a2%=W%:an$=" " : REMember set up everything.
6010 rep r1:REM set up a continuous loop
6015 if be%>1:be%=0
6020 if be%:beep 1000,30
6025 if w%<a2% then w% = a2%: REM you could use a colon here
      instead of the "Then".
6030 CUR W%.L% : print""
6035 ent$ = INKEY$(-1)
6040 if ent$$=chr$(96):be% = be% + 1: end rep r1
```

Lets outline where we are so far. Line 6005 sets all variables to a known state. The variable AN\$ will contain the final input. Line 6010 starts a loop. lines 6015. 6020 and 6040 allow you to switch on, and off a bleep sequence. What that means is that a noise is made every time you type something. To switch the noise on, or off, you slowly press the key displaying the pound sign. The chr\$(96) in line 6040 = the pound sign key. Line 6030 positions the cursor to the correct position, and prints a star (like the video-printer on BBC grandstand). You can, in fact, change the star to any other printable character. Line 6035 halts the program and waits for a key to be pressed. When a key is pressed the value of the key is stored in the variable ENT\$. Finally line 6040 checks to see if the pound sign has been pressed, if not the procedure continues. If yes. the loop is repeated. and the bleep switched on/off.

```
6045 if ent$ = chr$(10) : end def
```

The above line is very Important. What it says is: if you press the ENTER key, end the input routine. The information typed will be stored in the variable AN\$. In most other forms of basic the enter key is chr\$(13).

```
6050 if ent$ = chr$(192) then r3 : end rep r1
```

Line 6050 checks to see if the key with the arrow facing to the left has been pressed (next to CTRL). If it has, go to procedure R3. R3 allows you to delete a character.

```
6055 if len(an$) = limit% then L% = a1%: end rep r1
```

Line 6055 checks to see if you have typed the maximum number of characters. The word LEN checks the length of the variable AN\$. If the maximum has been reached nothing is printed, and the loop is continued.

```
6060 if bignum%+ L: r4
```

The above checks to see if you wish to type in numbers or letters. If bignum% = 1, you are only allowed to type in the characters specified by procedure r4.

```
6065 if big% = 1: end rep r1
```

```
6070 rem check to see if a number has been typed.
```

```
6075 if bignum% = 0 and not ent$ instr "QWERTYUIOPASDFGHJ  
KLZXCVCBVM-+());,./" :Ent$ = " ": end rep r1
```

Line 6075 appears complex. but it isn't. All it does is check 2 things. If the variable Bignum% = 1 you can only type numbers. If bignum% = 0 you can type letters. So. the first thing to be checked is if Bignum% = 0. If it does, the program checks if the key pressed is one of those enclosed between the inverted comma's of the INSTR function. If it is, the routine accepts it. If not, nothing is printed. The above is only an example to outline what can be achieved. You can if you wish, have numbers and letters accepted. To do so, simply incorporate the numbers within the INSTR FUNCTION.

```
6080 r2:REM reposition cursor. update variable AN$
```

```
6085 end rep r1:REM continue loop
```

```
6090 end def
```

That's the main routine. Now to finish things off.

```
7000 def proc r2
7005 if a% =limit% then end define
7010 CUR W%,L% : print ent$ : a% = a% + 1 : W% = W% + 6 :
an$ = an$ & ent$ : end def
```

Procedure r2 positions the cursor, prints on the screen (if printable) the value of the key last pressed, moves the cursor forward, and finally adds the value of ENT\$ (last key pressed) to AN\$. It then returns to procedure INFO for further input.

```
8000 def proc r3
8005 a% = a% - 1 : if a% < 1 then a% = 0
8010 if a% = 0 then an$ = " " : else an$ = an$(1 to a%)
8015 w% = w% - 6 : L% = a1% : if W% < a2% then end def
8020 if a% + 1 < limit% : CUR W%, L% ; print " _ " : else : CUR
W%,L% : print " _ "
8025 end def
```

Procedure r3 allows you to delete a character. It checks how many characters have been typed, repositions the cursor (deletes the character), and prints an underscore.

Before you run the input routine you can print a row of underscores " \_ ", to indicate the length of the input field (position them using CUR), then when you delete a character r3 will automatically delete the unwanted character and reprint an underscore.

```
9000 def proc r4
9005 if not ent$ instr "0123456789" : big% = 1 : ent$="" : end def
9010 big% = 0
9015 end def
```

Procedure r4 is a check to see if the numbers 0 to 9 have been granted. You can of course add other characters.

To call the input routine you must first decide where on the screen you wish to place the data, the number of characters you wish to accept and the type of data. i.e. numeric, string, or alphanumeric. You then simply call the procedure e.g.

Position at 30.30 with string input allowing 20 characters to be typed. The calling sequence would be as follows:

```
Bignum% = 0 : info 30,30.20
```

You would of course have to put the above in your program at a suitable point.

If you want more of the same sort of thing let me know. Also, if I can help anyone with progressing problems they can always telephone.

Frank Hope,  
15 Ash Close,  
Thorney,  
Peterborough PE6 0QS.  
(0733) 270477

### LONG STRINGS

Although another member has written about the long string problem with SuperBASIC, the following might also be of interest. as the technique is slightly different.

The solution, is quite simple - Just use INKEY\$ and the string concatenation, operator, '&'. The following program treatment illustrates the technique:-

```
long_string$ = " "  
REPEAT get_string  
    K$ = INKEY$(-1)  
    IF CODE(K$) = 10 THEN EXIT get_string  
    PRINT K$;  
    long_string$ = long_string$ & K$  
END REPEAT get_string  
:  
PRINT  
PRINT long_string$
```

Leon Heller

### USING TALENT'S ZKUL WITH DISKS

There are two files on the cartridge. namely Boot and Zkul. Boot consists of:-

```
100 c = RESPR(85400)  
120 ad = 176736  
130 LBYTES mdv1_Zkul , ad  
140 CALL ad
```

I was unable to load from the Microdrive - It was still loading after 20 minutes!

Therefore. COPY the files onto disk, but modify Boot program line 130 to

```
130 LBYTES flp1_Zkul,ad
```

Run boot from the disk but keep Talent's cartridge in Microdrive 1. The Microdrive will start up and the program will load in about 20 seconds

.  
Simon Callaghan,  
2 Heath Hey,  
Woolton,  
Liverpool. L25 4TF.

## BOOK REVIEW

### DESK-TOP COMPUTING WITH THE SINCLAIR QL for professions and businesses - by Barry Miles.

If you ever wanted a book that tells you all you ever wanted to know about the QL, but were afraid to ask - this is not it. It is part of the "The Sinclair QL Series", published by Hutchinson, with a foreword by Nigel Searle of Sinclair Research. Actually it is the same foreword in each book of the series. According to the preface this is a book designed to "help a new owner explore all the possibilities of the Sinclair QL as a desk-top computer". In my opinion however it is but a re-write of the QL User Guide which comes for free with the computer. Much to my dismay I was unable to extract from it any useful information not contained in said manual.

The book is divided into five chapters, the first being entitled "Getting Started" and in which one learns such useful things like the QL. "is very attractive to look at". You are then shown how to connect the QL... To be fair in this section at least the importance of making backup copies is stressed, even this should normally already have been inferred from the urgency with which the manual talks about making backup copies of the Psion software.

The next four sections take you through the four Psion packages, starting with Quill, going on to Abacus, followed by Archive and finishing by Easel. I found that none of these sections contained information which could not be unearthed by a careful read of the manual, even though it was perhaps presented in a lighter and slightly more accessible style than the somewhat dry tone of the user guide. Finally there is an annexe called "Sinclair QL SuperBASIC Keywords", lifted straight out of the manual. It has the same presentation as the "Keywords" section of the manual : first the name of the keyword then a slight explanation of what it does (but not more than the manual) and the the "syntax", followed sometimes by a comment. Thus the comment for the "beep" command is: "The SuperBASIC command is best used experimentally than syntactically". Sounds familiar, huh?

All in all I think I should have spent my £6.95 better on something else. The blurb on the book's jacket says that it is recommended by the QL User magazine. I can only hope that it means the QL User which now has culled publication and not the one in which our chairman is a regular contributor.

Wolfgang Lenerz

*/\*s I an sorry to say that the Hutchinson books which are uniformly dreadful, are recommended by EMAP's QL User magazine, which I write for! I have just heard that Century Communications, who publish some quite good books on the QL (plus my book on the 68000!) have merged with Hutchinson. \*/*

## A GRAPHICS QL

The founder of our local croup. Ron den Breems. has written the following demonstration program. It draws a QL, blinks the mdv lights. and recolours the image a few times, Then it clears the screen and prints in his characters the name and address of the group.

AW den Hartog,  
Tienvoet 2,  
3261 TP Oudbeyerland,  
The Netherlands.

```
.10 REPeat p : prog : END REPeat p
11 DEFine PROCedure prog
12 WINDOW 512,256,0,0 : SCALE 35,-2,0 : PAPER 0 : INK 0 : CSIZE 0,0 : CLS
13 BLOCK 511,255,0,0,4 : INK 0
14 FILL 1 : LINE 0,10 TO 47,10 TO 47,24 TO 0,24 TO 0,10 : FILL 0
15 FILL 1 : INK 7,0,1 : LINE 46,11 TO 48,11 TO 48,23 TO 46,23 TO 46,11 : FILL 0
16 y1 = 23.7 : y2 = 24 : INK 4
17 FOR x = 34 TO 45 STEP .5
18 FILL 1 : LINE x,y1 TO x+.25,y1 TO x+.25,y2 TO x,y2 TO x,y1 : FILL 0
19 END FOR x
20 INK 7
21 FOR y=11 TO 23 STEP 2
22 LINE 0,y TO 46,y
23 END FOR y
24 LINE 1,11 TO 1,21 : LINE 3,11 TO 3,21
25 LINE 3.5,11 TO 3.5,21 : LINE 33.5,10 TO 33.5,23
26 LINE 32.5,11 TO 32.5,19
27 x = 1
28 FOR y = 11 TO 19 STEP 2 : t
29 x1 = 3.5 : x2 = 32.5 : y = 19 : rij
30 x1 = 6.5 : x2 = 29.5 : y = 17 : rij
31 INK 0 : LINE 30.5,17 TO 32.5,17 : INK 7
32 x1 = 6.9 : x2 = 27.9 : y = 15 : rij
33 x1 = 7.9 : x2 = 25.9 : y = 13 : rij
34 x1 = 6.9 : x2 = 8.9 : y = 11 : rij
35 x1 = 24.9 : x2 = 26.9 : rij
36 l = 3.5 : x = 3.5 : y = 15 : lt : y = 11 : lt : x = 29 : lt
37 l = 4.5 : x = 3.5 : y = 13 : lt : x = 28 : lt
38 l = 3 : x = 3.5 : y = 17 : lt
39 ARC 31.2,15.3 TO 32.2,16,1,8 : ARC 29.6,15.3 TO 29.6,16.7,-2
40 ARC 30.8,18 TO 32.2,18,-3 : LINE 29.6,15.3 TO 31.2,15.3
41 LINE 29.6,16.7 TO 30.8,16.7 TO 30.8,18 : LINE 32.2,16 TO 32.2,18
42 AT 9,5 : PRINT "SINCLAIR" : AT 15,76 : PRINT "QL"
43 BLOCK 3,5,30,177,7
44 BLOCK 38,5,370,176,7,0,1 : BLOCK 38,5,430,176,7,0,1
45 FOR: x = 1 TO 5
46 BLOCK 3,5,360,177,7 : BLOCK 3,5,420,177,0 : PAUSE 20
47 BLOCK 3,5,360,177,0 : BLOCK 3,5,420,177,7 : PAUSE 20
48 END FOR x
49 FOR x = 1 TO 4 : RECOL 7,0,0,0,0,0,0
50 PAPER 0 : INK 7 : CSIZE 3,1 : CLS : PR1NT "      SIN_QL_AIR"
51 PRINT\ "      QL-USER GROUP"
52 PRINT\ "      KROONSTADDEEF 27"
53 PRINT\ "      3067 RT ROTTERDAM"
54 PRINT\ "      TEL. 010 - 551234"
55 PAUSE 500
56 END. DEFine prog
57 DEFine PROCedure rij
58 LINE x1,y TO x1,y+2
59 FOR x = x1 TO x2 STEP 2
60 t
61 LINE x+2,y TO x+2,y+2
62 END FOR x
63 END DEFine rij
64 DEFine PROCedure t
65 ARC x+.6,y+1.6 TO x+1.4,y+1.6,5,1
66 LINE x+.6,y+1.6 TO x+1.4,y+1.6
67 END DEFine t
68 DEFine PROCedure lt
69 LINE x+.6,y+.3 TO x+1-.6,y+.3
70 LINE x+.6,y+1.7 TO x+1-.6,y+1.7
71 ARC x+.6,y+.3 TO x+.6,y+1.7,-2
72 ARC x+1-.6,y+.3 TO x+1-.6,y+1.7,2
73 END DEFine lt
```

## THE GST QC COMPILER

When the QL was launched, a C compiler was promised. Over a year later, Sinclair have apparently abandoned the idea (Metacomco will now be selling Lattice C for the QL, which Sinclair were originally intending to do) and it is GST, of 68K/OS fame, who are first in the field with their QC compiler.

QC is not a full implementation of the language, being based on the public domain 8080 Small C compiler originally written by Ron Cain and published in Doctor Dobb's Journal of Computer Callisthenics and Orthodontia, with enhancements by J. Hendrix. I must confess to some involvement with this project, since I gave GST the source code for the original compiler. Howard Chalkley of GST first got the compiler working on a Superbrain CP/M system, producing code for the 8080 processor. He then modified the code generator to produce 68000 code. The resultant cross-compiler was then used to compile itself, and the object code ported on to a QL, together with the source code, and used to compile itself again, to produce a compiler that ran on the QL and produced 68000 code. A run-time package was then written. This process sounds quite easy, but it took Howard several weeks to get it all working. The end result of this incestuous process was the QC compiler.

The QC package comprises two Microdrive cartridges, containing the compiler, an assembler, and a linker, together with a comprehensive library. A book on C called, logically, "A book on C" is provided (I don't think much of the book, by the way), and a large manual. An editor is not supplied, as GST reckon that most users interested in C will already have one of the editor/assembler packages. I'd recommend Tony Tebby's editor (#1 from the library), if you don't have one already. The compiler is much easier to use if you have the QL Toolkit, but it is not essential.

Like most C compilers running on micros, QC compiles to assembly language which is then put through an assembler. The resultant relocatable object code is then linked with modules from the library, and any user-written libraries, to produce a binary file that may be EXECed in the usual manner.

GST gave me a beta-test copy of the compiler at the Swindon workshop, and I have been using it for a couple of weeks. As I have indicated, it compiles a subset of the C language as described in "The C Programming Language", by Kernighan and Ritchie, lacking floating point, structures, bit fields, and several other features, but it is ideal for systems programming, and learning the language. The beta-test copy had several sins of omission and commission, but these should have been fixed when the final version is unveiled at the Cambridge workshop on the 31st of March. The run-time library provides access to virtually all the features of QDOS available to the assembly language programmer, so that it will rarely be necessary to use the in-line assembly language feature provided.

QC will shortly be available at a price of £39.95, less 10% discount to IQLUG members. A full review will be in the next issue.

Leon Heller

## FOR SALE

Two QLs (Version JM) at £300 each. Phone Don Forbes on 01-688 5794.



### A SIMPLE PROGRAM USING THE QC COMPILER

The following file listing program, written in C for the GST QC compiler, should illustrate the flavour of the language. The program gets the file name from the command line. Thus, typing

```
EW MORE_BIN;"FLP1_MORE_C"
```

would cause the program to list its own source code until the window is full (17 lines, since the default QC window has room for 18 lines), and then wait until any key is pressed. then display the next 17 lines, and so on.

```
# include<stdio.h>

# define LINES 17 /* default QC window has 18 lines */
# define ESCAPE 27 /* code for escape key */

/* global variables */
int c, n1;

main(argc, argv) int argc, *argv:{

    int chan;
    char filename[20];

    getarg(1, filename, 20, argc, argv); /* get filename from command line*/

    chan = fopen(filename, "r"); /* open file for read */
    if (chan == 0) {
        Puts("File not found");
        exit(0);
    }
    n1 = 0;
    while ((c = getc(chan)) != EOF){
        putchar(c);
        More();
    }
}

More(){ /* increment line counter and check for window full */

    if (c == '\n') n1++; /* increment line counter if EOL */
    if (n1 == LINES) cmd(); /* if window full wait for key depression */
}

Cmd(){ /* wait for key depression, quit if <ESCAPE> */

    int key;

    N1 = 0;
    key = 0;
    while (key == 0) key = poll(0);
    if (key == ESCAPE) exit(0);
    cls(0);
}

}
```

Leon Heller

## CROYDON SUB-GROUP

The Croydon sub-group is not just for people in the Croydon area, but is open to anyone. Please contact me on 01-688 5794 for further information.

Don Forbes.

### **“QL SUPERBasic - A Programmers Guide” by John Wilson Published by MICROPRESS at £6.95 ISBN 0-7447-0020-5**

John Wilson has written an introductory book to complement the User Guide. He assumes that you have grasped at least the basics of programming and provides examples and utility procedures to assist understanding and to ease the programming task.

Initial chapters clarify the use and understanding of procedures, condition testing, and repetition with emphasis on what the different forms are and when best to use them. A chapter on Data Structures follows. Such a title is unusual in beginners books and presages well for the book as a whole. Chapters on arrays and strings follow and include functions for padding strings with blanks. Procedures are provided for printing instructions on the screen without breaking up words and for “PRINT USING” type printing.

“More Complex data structures” advises on record design, and provides functions/procedures for reading and writing data in arrays without knowing anything about the physical array details. The chapter goes on to stacks and queues with procedures to use them. However it does not indicate why you might want them and refers to linked lists and trees.

The QL Display chapter, really clarifies the effect and use of the WINDOW, STRIP, BLOCK, CURSOR, AT functions and their relation to the pixel and graphics coordinate systems. The effect of OVER and the RECOL command are clearly explained and a sample program given.

The Sound chapter includes a variant on the common BEEP evaluation, but also a tune playing program. Timing is also covered.

The filing chapter was a disappointment. Though backup and duplicate programs are provided, John is unaware of the EOF function and the automatic verification of saved files. A comprehensive file management program would have been more in tone with the book.

The penultimate chapter gives an example of how to design and structure a graphics program, and gives a 400 line program for drawing picture on the screen. There is a short index.

Compared to many others, this book is worth buying. I would prefer it to have expanded some topics, it contains very little waffle.

### **“Introduction to Simulation Techniques on the Sinclair QL” by John Cochrane published by SUNSHINE at E6.95 ISBN 0-946408-45-9**

Initial chapters tell what simulation is all about. Some uses of simulation on mainframes are mentioned but the reader is not told that purpose made languages such as CSL, GPSS and SIMULA exist..

Types of simulation, design and verification are dealt with, though describing Markov chains in terms of system state matrices is unlikely to enlighten those who started in ignorance.

An ABACUS cash flow projection is the first example. A simulation true, but one also found in books on ABACUS. A 200 line program simulating a four stroke petrol engine captures the true simulation flavour. A 280 line program to do critical path analysis (simulation of work activities) and a 300 line program based on Markov chains to predict the weather complete the practical examples.

The final chapter describes a real time simulation, but this turns out to be a 300 line volley-ball game program.

There is no index. There is a glossary and an entirely unnecessary 8 page chapter describing the QL and its 68008 chip.

Unfortunately real life simulations often require complex mathematics. Two classic simulation models concern the number, of supermarket checkouts or the number of dryers in a laundrette, needed to keep the queues to some defined size. Their analysis requires the equations of queuing theory and sampling from a Poisson distribution. Despite this, the inclusion of one of these models would have rounded out the book more and presented a practical aspect of simulation, which many readers may have been able to modify to their own real use.

**“Exploring the Sinclair QL” by Andrew Nelson.  
Published by INTERFACE £4.45 ISBN 0-907563-84-80**

This book is a rewrite of the Beginner's Guide with all the hard bits left out. Continual reference to the way the SPECTRUM does things, identifies the target audience.

The book starts with plugging the QL in and continues by introducing the ideas of variables, identifiers, operations, arrays and string slicing. You are advised to program as if in SPECTRUM BASIC and gradually introduce the advanced structuring features of SUPERBasic!

The chapters on graphics, printing and colour add nothing to the User's manual and would not leave the reader able to handle the positioning problems resulting from use of AT, CURSOR, CSIZE and PRINT.

The chapter on dealing with data stops at READ/DATA and RESTORE. No examples of using OPEN/CLOSE or any file operations are covered. In describing INKEY\$ the SPECTRUM usage of 10 IF INKEY\$ = "" GOTO 10 is described in preference to the QL approach of INKEY\$(-1).

A chapter on QDOS gives memory maps and is quite out of context with the simple approach of the remainder of the book. Meanwhile many keywords such as COPY and DIR are merely listed but not described.

Too many errors are present, possibly from pre-release documentation. Reference is made to WHEN ERROR, CAT, TRACE and default device names. We are told that LOAD pgm1 and SAVE pgm2, will act on mdv1\_.

There is no index, but the 4 page table of contents is very helpful.

This goes at a much slower pace than the official Guide, but even at the price, it leaves out too much that is essential.

Gerard Phelan,  
17 Gunnersbury Court,  
Bollo Lane,  
LONDON. W3 8JL

### **RUN-TIME ARCHIVE**

Members thinking of developing commercial software using Archive might be interested in the run-time version that is available from Psion. I don't know how much it costs. but it does provide an increase in available memory of about 35000 bytes. as opposed to the 20000 bytes available with the standard package, according to Charles Davies of Psion. The Quest Tally package uses the run-time version, by the way.

Leon Heller

### **NEW PRODUCTS**

TDI Ltd, of Bristol are advertising UCSD Pascal for the QL. At just under £t100. I got UCSD Pascal for ma TRS-80 Model 1 some years ago but I didn't like it very much. The main problem was the operating system, which was rather primitive, and the limited arithmetic precision. FORTRAN 77 is also available from TDI, which means that the very nice PICASO Graphics package developed by John Vince at Middlesex Poly, could be put on the QL. I had a hand in putting this package on a CP/M system with a high res. colour graphics sub-system, and it would not be too difficult to sort it to the QL. A lot of the BBC2 Money programme graphics are done with PICASO, by the way.

Metacomco tell me that they hope to have Lattice C on the QL within a couple of months. Unlike GST's QC, Lattice C is a full implementation of the language. It will cost £89.95, which might sound expensive. but is peanuts compared to the £450 that the IBM PC or MSDOS version costs. Now that 0.5 Mb RAM expansions are available, they are thinking seriously about putting full Cambridge LISP on the QL. This LISP package includes both an interpreter and a compiler. I have seen it running on an Acorn ABC machine, and it was very impressive. Metacomco will soon have a full ISO Pascal compiler available. Apparently memory is a bit tight, and they have to use the screen memory when compiling, which makes for some interesting visual affects.

Expert Systems International will soon have PROLOG for the QL. PROLOG is supposed to be the language chosen by Japan Inc. for \_their fifth generation systems, but I don't believe it. Anyway, PROLOG is quite a nice language for developing intelligent data bases and expert systems.

Simplex Data Ltd. tell me that they will shortly be bringing out an adaptor, with power supply for £49.50, enabling the use of both their RAM expansion and a peripheral, such as a disk controller. at the same time. You won't be able to use two peripherals. However, as a bug in the AH and JM ROMs will not allow two peripheral devices on the bus together. Sinclair's expansion unit contains its own ROM which Gets over this problem. and future ROMs will have the bug fixed.

Leon Heller

### **BUGS IN THE BASIC**

I don't like to say it in case Sinclair may hear but I'm getting quite fond of SuperBASIC. I recently re-wrote a program written in a bog standard, one-statement per-line form of Basic with it's inevitably contorted GOTOs and found the SuperBASIC form so readable and compact that I almost forgave Sinclair for their sins.

However, most of my efforts in QL programming over the past seven months have been directed to converting programs in structural and civil engineering which I

originally wrote on Commodore machines and, In the process, I have found from painful experience that SuperBASIC has flaws which, if known in advance, may be less painful and I hope I can save others some of the tedium I have suffered in proving that what one would assume was a failure of the programmer was actually a failure of SuperBASIC.

1. The dreaded PRINT epidemic.

You are happily programming along adding code and RUNning it every now and then to ensure it does what is intended and, of course, RESAVEing it frequently then, without warning, the next RUN fetches up with a BAD NAME or ERROR IN EXPRESSION message in a line which has previously behaved immaculately. You list the line to find that where it used to contain a nice variable name it is now replaced with PRINT! If it's the first experience of this Sinclair Sabotage you stare in disbelief; but worse, you list the program to find that the particular variable, and perhaps several others, have been replaced throughout your program by this obnoxious word PRINT. If you did the 'wise' thing you RESAVEd before you ran it so now you have no option but a full re-edit. Don't ask me why it happens, I can only say that when I told Tony Tebby he was very interested but apparently not at all surprised.

2. Loop counters as parameters.

It is often convenient for the counter used in a FOR loop in one PROCEDURE to be passed as a parameter to another PROCEDURE. I have repeatedly, but not invariably, found that the loop in the calling PROCEDURE will only operate once because the counter somehow gets incremented to the terminal value of the FOR statement. The problem never arises if the counter is used in a bracket of an array value but in a PROCEDURE call where the counter is used as a numeric parameter it often does. I now assign a second variable to the value of the counter and use the second variable as the parameter which gets around the problem. Again I have no idea why this occurs.

If others have found these or similar problems I would like to hear from them and, of course, if anyone can provide an explanation I am I sure we would all like to know. I would have contacted Sinclair if I had thought they had anyone there to listen.

John Tanner, 43 Northumbria Drive, Bristol. BS9 4HL Tel:-0272-623401

**OEL "DOING A SINCLAIR"!**

**Q COM communications package from OEL Cumbria**

**Q COM interface - Q MOD modes - Q CALL auto dial/call unit**

Prospective buyers should be aware of two facts.

1. Although the package as advertised and orders invited before Christmas (subject to 28 days for delivery) the goods are NOT yet available for distribution (I write on 7<sup>th</sup> March 1985 after checking with OEL).

2. If you want to communicate with Bulletin Boards operating at 300 baud - and these are very popular - then Q MOD is not for you. It is restricted to 75/1200 or 1200/1200 baud. Mind you, you can still buy the £79.95 Q COM to interface your QL with a different modem, and OEL will very happily supply you from stock with a £175 Telemod 3 which will work with Q CON though not with Q CALL. But, if you do decide on this alternative, it might be best to wait until Q CON is actually available in the shops. £175 is a lot to pay for what amounts to an unlovely, dust-gathering object: virtually useless until the rest of the bits arrive ...

Francis Cameron, 12 Norreys Avenue, Oxford. OX1 4BB (0865 240058)

### FORTH ON THE QL

We have developed a version of FORTH-79 for the QL, and have been extensively testing it. I knew that SuperBASIC was slow but I was surprised when I tried benchmarking FORTH and SuperBASIC using a Sieve of Eratosthenes program, listings included with this letter. The FORTH version took less than 5 seconds and the SuperBASIC version 212 seconds (using integer variables makes it even slower ! ). I should be very interested to hear how other languages, such as PASCAL, on the QL compare with these times. We are going to market this version of FORTH which fully implements the FORTH-79 Standard with Double-Number extensions ( i.e. 32 bit integer as well as 16 bit Integer working ). Other features include access to QDOS traps and vectored utilities, a full screen editor for standard block generation, operation with named files as well as the standard FORTH blocks and redirectable I/O. If anybody is interested please send 15 pounds to R.E.Jackson at the attached address.

On another subject, why is everybody so rude about QUILL etc, after all, PSION did warn us with the name, a quill is a slow old fashioned writing tool, liable to blot your work, and has long been superseded by more modern tools. When is the next version meant to be released?

Has my QL got a fault? Pressing CTRL F5 doesn't do anything, according to the manual it should give the code 249.

```
90 REMark Sieve of Eratosthenes
100 size = 8190
110 DIM flags(size)
120 FOR i = 0 TO size
130     flags(i) = 1
140 NEXT i
150 count = 0
160 FOR i = 0 TO size
170     IF flags(i) = 1 THEN
180         count = count + 1
190         prime = i + i + 3
200         IF prime < 128 THEN
210             FOR j = prime + 1 TO size STEP prime
220                 flags(j) = 0
240             NEXT j
250         END IF
260     END IF
270 NEXT i
280 PRINT count; " primes"
290 STOP
```

(Sieve of Eratosthenes - Forthwrite Vol 3 No 2)

```
8190 CONSTANT SIZE
VARIABLE FLAGS SIZE ALLOT
: FINDPRIMES3
  FLAGS SIZE 1 FILL
  0 SIZE 0
  DO FLAGS I + C@
    IF 1+ I DUP + 3 + ( DUP . ) DUP 128 <
      IF SIZE OVER I +
        DO 0 I FLAGS + C! DUP +LOOP
      THEN DROP
    THEN
  LOOP . ." Primes" ;
```

Gerald Jackson, 15 EDGEHILL, LLANFRECHFA, CWMBRAN, GWENT> NP44 8UA.

### **CAMBRIDGE WORKSHOP**

The Cambridge workshop will be held at the University Arms Hotel, Regent Street, Cambridge. on Sunday 31 March. An interesting new peripheral might receive its first public showing at the meeting. Details have already gone out to members in the area. but anyone else who would like to come along will be very welcome. Give Brian Pain a ring if you would like more information.

Leon Heller

### **ZX MICROFAIR**

As at the previous Microfair, last November, IQLUG had a stand at the Alexandra Palace on Saturday 9 February. We didn't get very many new members, probably because the atrocious weather kept a lot of people away, but it was nice to see those existing members who braved the elements and turned up.

CST/Computamate seemed to be doing very good business with their disk interface and disk drives, especially as they were giving quite sizeable discounts to people who purchased their goodies at the show.

Medic also had a large stand which featured their disk interface and drives. Unfortunately, their hardware was on a van which crashed en-route, and didn't materialise until after lunch. They have promised me one of their units, which will be reviewed in a supplement to this issue, if they are able to get it to me in time. Their prices look extremely competitive. Several IQLUG members were helping out on the Medic stand, and I was surprised to see one member, who arrived wearing a T-shirt, suddenly become transmogrified, especially for the occasion, by the addition of a natty white shirt and tie! Chris Skoglund of Medic, managed to talk Sinclair into letting him try the C5 which was featured on their stand, when the show ended. This was the first time I have actually seen anyone driving one of the things - a truly awesome sight, especially when, driven flat-out by Chris.

Eidersoft, who had the stand next to ours, had several items for the QL, such as a drawing program called QLArt, and packaged Archive procedures. They were responsible for making everyone in the vicinity jump when their stand collapsed, nearly causing some damage to CST personnel and their wares.

Sinclair seemed to be doing a roaring trade in Spectrum upgrade kits, and several of our members who also have Spectrums were snapping these up so that they could have the same funny keyboard on both machines.

To cap it all, my car wouldn't start when I tried to go home. Luckily, there were several others in the same boat, and we managed to find some jump leads and eventually got all our cars going again.

Leon Heller

### **PSION CHESS COPY PROTECTION**

One or two members have found that it is remarkably easy to clone the PSION chess master cartridge. No, I won't tell you how to do it. I told Sinclair and Psion, about this, and Psion admitted that the technique would work, but that the protection would, effectively, prevent large-scale piracy.

Leon Heller

**CAPS LOCK ON/OFF**

Herewith is my Machine code program to enable to you switch Caps Lock on and off in a BASIC Program. If you wish you may Link in the code into a machine code program.

PAGE 1      QL    MACRO ASSEMBLER VERSION 1.7    CAPS LOCK ON/OFF

LOC	OBJECT	STMT	SOURCE STATEMENT
2			
3	*		Caps Lock Utility Program
4	*		Using the Metacomco Assembler.
5	*		By J. W. Robinson.
6	*		10th February 1985.
7			
8	*		Program is Position Independent.
9	*		It requires 72 bytes of memory.
10			
11	BP_INIT	EQU	\$110
12			
13		LEA	PROC_DEF,A1            Point to Definition Block.
14		MOVE.W	BP_INIT,A2            Point to Linker Vector.
15		JSR	(A2)            Call Basic Linker Routine.
16		RTS	
17			
18	PROC_DEF	DC.W	2            No of Procedures to be Added to Basic.
19		DC.W	CAPS_ON-*            Address of first Procedure.
20		DC.B	7,"CAPS_ON"            Name of first procedure.
21		DC.W	CAPS_OFF-*            Address of second Procedure.
22		DC.B	8,"CAPS_OFF"            Name of second Procedure.
23		DC.W	0            End of Procedures.
24		DC.W	0            No of Functions.
25		DC.W	0            End of Procedures.
26			
27	CAPS_ON	MOVE.L	*\$FF000000. (\$28088)            Switch on Caps lock.
28		MOVEQ	#0,D0            Clear D0 so no Errors
			Returned to Basic.
29		RTS	Return to Basic.
30			
31			
32	CAPS_OFF	MOVE.L	#0,(\$28088)            Switch off Caps lock.
33		MOVE.Q	#0,D0            Clear D0 so no Errors
			Returned to Basic.
34		RTS	Return to Basic.
35			
36		END	

J. W. ROBINSON,  
37 MARGARITE WAY,  
WICKFORD,  
ESSEX. SS12 0ES

**A QUILL MERGE PROGRAM**

I am thinking of writing a program to verge details from an Archive file with a Quill document, for doing mail shots and so on. Would any members who would be interested in such a program please get in touch with me, preferably by Phone on (0908) 613004. The program will be written, in C, using the new QC compiler from GST

Leon Heller



### How long is a string?

Richard Kingslake (QUANTA February 1985) is quite correct in that Inputting long strings (greater- than LEN = 127 is not on because the buffer gets full.

However, strings can be concatenated together, much longer than this, and printed to the screen without trouble. If long string input is actually needed then...

```
100 LET a$ = "": b$ = ""
110 REPEAT string_in
120     INPUT a$
130     IF a$ = something or other THEN EXIT string_in
140     LET b$ = b$ & a$
150 END REPEAT string_in
```

...will serve, the long input ending up quite happily as b\$

Reading-in long strings from Microdrive files presents the same problem, as the buffer quickly gets full and refuses to accept any input. Which was how I discovered this particular feature.

The following program, in which a\$ was subsequently quadrupled in length (to LEN = -736), worked O.K. as characters were printed to Microdrive, and then read back, one at a time.

```
200 CLS
210 INPUT a$
220 OPEN NEW #5, mdv1_string
230 FOR j =1 TO LEN(a$)
240     PRINT #5, a$(j)
250 END FOR j
260 CLOSE #5
270 INPUT z$ : CLS : b$ ="" (This gave me a break before reading the file back in.)
280 OPEN_IN #6,mdv1_string
290 REPEAT reading
300 IF EOF (#6) THEN EXIT reading
310 INPUT #6, i$
320 b$ = b$ & i$
330 END REPEAT reading
340 CLOSE #6,
350 PRINT b$
```

The way the QL stores the string can be illustrated by the command - COPY MDV1\_string TO SCR\_ as the characters get printed in a vertical line.

This process is slowish and uses unnecessary memory (lots of newline characters) in long strings. It can be speeded up, and memory saved, by the following changes...

```
230 FOR j = 1 to LEN(aS) STEP 128
240     PRINT #5, a$(j TO j+127)
```

...and the input read back as i\$ in line 310 comes in 127 character chunks (plus newline, of course). It COPYs out like this, too. For the particular string I used to test the program (LEN = 736) it used two Microdrive-cartridge Sectors, compared with three for character-by-character saving.

Having typed all that, I read in QL Advanced User Guide (Adrian Dickens) that console I/O can be specified by CON\_WxHaXxY\_K where K is the number of characters in the keyboard type-ahead buffer. That's it, I thought, quickly specifying a buffer of 256 bytes. But it isn't - no effect that I could discern. Perhaps someone out there can explain.

Nice one, Mr. Kingslake.

My thanks to Ken Huber (same issue) for the \_exp to Quill tip. It's worked nicely here.

**My thanks also to Dr. Basil Lee (ibid) for pointing out what now seems obvious, i.e. that one can redefine function keys in QUILL to give other fonts. QUILL thinks its just underlining this paragraph, but as you can see (I hope) it's printed with proportional spacing.**

And now its not - another nice one.

Ron Humpnries,  
16, Coniston Road,  
Sutton Coldfield. B74 3LE.

#### **QUILL VERSION 2 BUG SHOCK HORROR!**

20 minutes after receiving my Psion software upgrades I had found a serious bug in Quill. I was using it on my CST disk-based system and found that merging a large document with another one resulted in an 'out of memory'. Pressing <SPACE> as instructed, "7333 words" was displayed. From then on it was impossible to save the document - "file I/O incomplete" kept coming up - and Pressing ESCAPE, to get back into normal mode. which should have enabled some of the document to be deleted. and the remainder saved, zapped the entire document. leaving me in Quill staring at a blak screen and the document irretrievably lost. The same thing happened with Microdrives and the Simplex 256K RAM expansion (although the limit was much higher. about 11.000 words), so it was nothing to do with the CST disk system. I have informed both Sinclair and Psion of this problem. and Psion admit that Quill's handling of an out of memory condition is at fault. in that there is not enough memory left to enable recovery to take place. Martin Baines of CST tells me that he found this bug in a beta-test copy of Quill Version 2 some months ago. and told Psion about it. This reflects rather badly on both Psion and Sinclair. as they are either incompetent, or trying to get yet more money out of users for further upgrades

I have just been testing Archive Version 2. and found that it has a similar bug which is just as disastrous.

Leon Heller

#### **ALCC LONDON COMPUTER FAIR**

The ALCC London Computer Fair takes place at the Central Hall, Westminster. on 18 - 20 April. We will be sharing a stand with CST. and need volunteers to help man (or woman) the stand. You will get free entrance to the show, of course, and plenty of time to look around. Please give me a ring if you can help.

Leon Heller

### BASIC EXTENSIONS.

One of the most useful features of the Sinclair QL microcomputer is the ability to mould the capabilities of the machine to the user's requirements. The In-built language of the machine is a version of BASIC known modestly as SuperBASIC, is built in such a way as to have a central core of code surrounded by a number of extensions which go together to make up the 'flavour' of the language. Most of the commands, although not all, are implemented from a linked list of subroutines known as Procedures and Functions. This structure allows the user to define extensions to the language in either BASIC or machine code by adding to the linked lists. This is in fact quite possible in SuperBASIC and even seems to be encouraged for users of both SuperBASIC and assembly language. These extensions then become new commands which may be accessed in the same manner as already implemented commands.

Command extensions written in SuperBASIC only exist in the machine while the DEFINED PROCEDURE or DEFINED FUNCTION code actually exists within the current program and therefore should be considered to be only temporary extensions to the language. In other words, changing programs in SuperBASIC will cause the loss of all PROCEDURES and FUNCTIONS defined within that program. They will be replaced by any DEFINITIONS provided in the program which next comes into memory.

Command extensions written in assembly language may be loaded into memory and linked into the machine in such a manner as to become a permanent part of SuperBASIC, at least until the machine is RESET. This ability allows us to add extensions, in the form of new commands, to both increase the functionality of SuperBASIC and to significantly increase the performance of SuperBASIC programs. It becomes quite feasible to write commonly used segments of code as assembly language subroutines and merely name them as command PROCEDURES. Efficiently written assembly language PROCEDURES can perform many times faster than the equivalent SuperBASIC code and it therefore is to our own advantage if we utilise this fact.

The manner in which assembly language PROCEDURES and FUNCTIONS are implemented is fully documented in the book 'QL Advanced User Guide' by Adrian Dickens, published by Adder. Quite nimbly, this involves writing an assembly language program with the following code at the start of the program:-

bp_init	equ	\$110	Procedure initialisation.
*			
Start	lea.l	proc_def,a1	Pointer to definitions
	move.w	bp_init,a2	Initialisation subroutine
	jsr	(a2)	perform initialisation
	Moveq	#0,d0	no errors
	Rts		Return to SuperBASIC

This routine will perform the required linking of all defined PROCEDURES and FUNCTIONS contained in the current module. In order to define the actual pieces of code themselves, it is necessary to provide a list of all the PROCEDURES and FUNCTIONS actually coded. This is done as follows:-

```
*
*
* Firstly we must provide a list of all the PROCEDURES provided by this code unit.
* The number could well be zero, but we will define two PROCEDURES and one
* FUNCTION.
*
proc_def    dc.w    2                number of procedures
*          this may be zero if none
p1         de.w    clck_cl-p1       offset to PROCEDURE code
          Dc.b    7,'CLCK_CL'      number of bytes in name,
*          followed by the name
*
*
```

```

P2          dc.w    clk_st-p2          offset
           dc.b    7,'CLCK_CL"        number, name
*
           dc.w 0                      end of PROCedures marker
*
* Following the PROCedures, we must define the FUNctions. As before we are at liberty to define
zero but must at least define the whole table with start and end markers
*
           dc.w    1                      1 FUNction
           dc.w    clk_rd-f1            offset to FUNction
           dc.b    7,'CLCK_RD"        number, name
*
           dc.w    0                      end of FUNctions marker

```

The rest is simply a matter of writing the code required to fulfil the user's needs. Full details of how to do this and a lot of other very interesting things is also contained in Adrian's book.

I have written a set of SuperBASIC PROCedures and FUNctions for my own purposes, some of which merely allow access to QDOS functionality from SuperBASIC, others of which add greater flexibility to SuperBASIC itself. A description of some of the most interesting of these is provided below:-

CURSOR_ON #n	Allows the user to enable the cursor for a particular channel. This is useful if your program is using the SuperBASIC command INKEY\$ as this does not enable the cursor.
CURSOR_OFF #n	Allows the user to disable the cursor for a particular channel.
HEAP_AL(n)	Allocate memory from the common heap. This allows memory to be allocated in the required quantities at a fixed place in memory, SuperBASIC returns the address of the base of the allocated memory block.
HEAP_DA(base)	De-allocate memory back to the common heap. Return some memory that was obtained with HEAP_AL.
CLCK_CL #n	Clear a 50hz clock and start it counting. This is just one of 7 routines to handle 50hz clocks. These routines are very useful in providing timings of program runs and event count-downs. Clocks are accurate to 0.02 seconds provided that interrupts are not inhibited while a clock is running. PROCedures are provided to set, clear, start, stop, alarm and adjust a clock. A function is provided to read a clock value. It is possible to set more than one clock running at the same time and a maximum of 12 are provided.
PEEK_C(addr,n)	Perform a fast, continuous PEEK from memory (addr) to a string n bytes. This allows fast updating of the screen to be performed which would otherwise have to be handled entirely in assembly code.
POKE_C(a\$, addr)	Perform a fast, continuous POKE of memory from a string of n bytes. Used to move data in memory quickly.
ASC(addr,n)	Format n bytes of memory from absolute address (addr) to Ascii format. The returned string will contain the Ascii characters for the bytes read, provided that they are printable characters. All non-printable values will be returned as full stops.
DEC(addr,n)	Format n bytes of memory from absolute address (addr) to Decimal format. The returned string will contain the value of each byte as a 3

digit decimal number followed by a single space. E.g. We have in Memory the following data:-  
200000 ... C5 51 012 097 129  
and request n\$ = DEC(200000,5), the returned string n\$ will contain:-  
"005 051 012 097 129 "

HEX(addr,n)            Format n bytes of memory from absolute address (addr) to Hexadecimal Format. The returned string from the above example would be:-  
"05330C61812

SORT array,n            Sort an entire array into the sequence defined by n. The meaning of n will depend upon the type of array to be sorted. For a numeric array, either integer or floating point, +ve will signify ascending and -ve will signify descending. For a string array, the sort type is the same as the comparison type defined in the "Concepts" section of the QL User's Guide. It is also possible to specify a portion of an array to be sorted and to sort a single dimension of a multi-dimensioned array.

The last item is perhaps by far the most significant use of assembly language extensions to SuperBASIC. For anyone who does not believe it, try the following:-

```
100 DIM n(999) ..            create an array of 1000 FP no-s.  
110 FOR i = 1 TO 999 ..      loop for 1000 FP no's  
120 n(i) = RND(-32767 to 32767) .. fill the array with random no's  
130 NEXT i  
.....                      put your code here to sort the array into  
                            Ascending order
```

Time the sorting operation and you will see that if you use a fast sorting algorithm, the sort time would be of the order 250 seconds. Using the assembly language PROCEDURE it is possible to perform the same sort in less than 1.3 seconds, a significant saving, almost two orders of magnitude, in both time and convenience.

The extensions outlined above are -commercially available for the sum of £9.95 Pounds on your own Microdrive cartridge. As a special concession to IQLUB members, the extension, mentioned above plus others making a total of 29 PROCedures and 14 FUNctions, in all as a file handling utility, with listings and binary code are available for the sum of five pounds on your own cartridges. Send your Cartridge and five pounds (International Money Order/ Cheque to:-

V. R. Dwight.  
T:m1 Softrinio.,  
Suvikuja 3H14  
02120 Espoo 12  
Finland.

/\* Ron sent me a copy of his software, and it seems very good. The sort feature is worth £5 on its own, in my opinion. \*/

**GST Computer Systems Limited**

We offer a 10% discount on all QL products to IQLUG members.

<u>Hardware Products</u>	<u>Price</u>
Microdrive Tapes	£1.95
RS232 Printer Cables	£12.95
 <u>QDOS Software Products</u>	
QL Assembler	£39.95
QC Compiler	£59.95
Macro Assembler	Available soon!
 <u>68K/OS Products</u>	
68K/OS Operating System on Plug-in Rom Board	£99.95
68K/OS Compiler	£59.95
 <u>Books</u>	
Motorola 68000 Reference Manual	£8.95
Programming the 68000 (King 8 Knight)	£8.95

We will be at the IQLUG meeting in Cambridge on 31 March, or write to:  
GST Computer Systems Limited,  
The Green,  
Willingham,  
Cambridge. CB4 5JA  
Tel. Crafts Hill (0954) 81991

TUNES ON THE QL.

I thought that Ken Huber's tune (QUANTA p27 Feb 85) was really excellent and I very much enjoyed playing it. I selected from his list of evenly tempered notes the scales and played them several times too. If the figure for pitch were proportional to the frequency, the steps should have been 2:1 and the intervals would be the twelfth root of two. I suppose one could smooth out the curve by using:-

1000 S = 1/12 : p = 1247/57	2000 S = 1/12 : P = 57/24
1010 FOR N = 0 TO 12	2010 FOR N = 0 TO 12
1020 PRINT 57*P^(N*S)!" ";	2020 PRINT 24*P^(N*S)!" ";
1030 END FOR N	2030 END FOR N

To my ear (which admittedly isn't very reliable in this field) the scales selected from these sounded quite good. The twelve notes to each octave would be according to this interpolation:-

124,	116.2,	108.9,	102.1,	95.7,	89.7	
84.1,	78.8,	73.9,	69.2,	64.9,	60.8,	57
57,	53.0,	49.4,	45.9,	42.7,	39.7	
37.0,	34.4,	32.0,	29.8,	27.7,	25.8,	24

But it is doubtful whether this is any better than Ken's figures - one really needs to measure the frequency accurately to feel confident. One would guess that the tiny Beeper (piezoelectric!) would change in pitch with increasing temperature: if so, a lengthy Tune might end up at a different pitch and any time spent in measuring frequencies accurately would be wasted. The accuracy of the BEEP does not of course justify the use of the first place in decimals since the QL rounds off each figure to an integer and 55.49 is played as precisely the same pitch as 54.51: and what is proportionately worse, 23.51 is the same as 24.49. I am sure that the word BEEP was chosen advisedly!

I was interested to see (and hear!) on TV a report on the work at Lancaster University's Music Department using a BBC computer. A sequence of music lessons of a 'Do It Yourself' kind have been made, naturally with full graphics. I am hoping that this type of educational programme will be written for the QL, together with appropriate interfaces to hardware.

Bill Cowhig

#### **DO YOU WANT AN INDEX FOR YOUR QL MANUAL**

If so one will be available  
early in April from:—  
P. J. HEDGES,  
1 Cirkstead Road,  
Bury St. Edmunds,  
Suffolk IP33 2EW

Two versions are available:-

- A). based on the Original QL  
and Psion Manuals.
- B). Based on the Original QL  
and the updated (12/84)  
Psion manuals.

Please state clearly which  
version you require.

Total Cost: £1 (inc postage)

*Hungry Harry  
in  
Haunted House*

Evade the ghosts and bats as you guide Harry to the garlic. Kill Dracula by sunset or he will bite your neck.

- FULLY MACHINE CODE GRAPHICS
- MULTILEVEL ARCADE ACTION
- CONTINUOUS HIGH SCORE TABLE
- COLOURFUL GRAPHICS

GREAT FUN FOR ALL THE FAMILY.

SPECIAL PRICE FOR QUANTA MEMBERS

*£ 7.99*

inc p&p

DISCOVER THE TRUE POTENTIAL OF YOUR QL WITH THE GREAT PROGRAM

*68000* MACHINE CODE TUTOR

A SIMPLE, WELL DOCUMENTED BEGINNERS AID TO THE TRICKY PROCESS OF MACHINE CODE PROGRAMMING

*£ 7.99*  
INC P&P

CREATE YOUR VERY OWN MULTICOLOURED MULTISIZED CARTOON TYPE CHARACTERS



*"Sprite Designer"*

This easy to use graphics package makes it easy to produce fantastic characters for use in your own programs.

Package includes a procedure to include in your programs for quick sprite movement

*£ 7.99*  
INC P&P

SPECIAL OFFER FOR QUANTA £16.99 FOR 3 PROGRAMS

ALL CHECKS TO BE MADE PAYABLE TO SNOWSOFT AND SENT TO  
6 BOUSFIELD CRES., NEWTON AYCIFFE, CO. DURHAM

• SNOWSOFT • • SNOWSOFT • • SNOWSOFT • • SNOWSOFT • • SNOWSOFT •